Unsupervised Anomaly Detection in Multi-Topic Short-Text Corpora

Mira Ait-Saada[1,2] and **Mohamed Nadif**[1]
[1]Centre Borelli, Université Paris Cité, 75006, Paris
[2]Caisse des Dépôts et Consignations, 75013, Paris
{mira.ait-saada,mohamed.nadif}@u-paris.fr

Abstract

Unsupervised anomaly detection seeks to identify deviant data samples in a dataset without using labels and constitutes a challenging task, particularly when the majority class is heterogeneous. This paper addresses this topic for textual data and aims to determine whether a text sample is an outlier within a potentially multi-topic corpus. To this end, it is crucial to grasp the semantic aspects of words, particularly when dealing with short texts, since it is difficult to syntactically discriminate data samples based only on a few words. Thereby we make use of word embeddings to represent each sample by a dense vector, efficiently capturing the underlying semantics. Then, we rely on the Mixture Model approach to detect which samples deviate the most from the underlying distributions of the corpus. Experiments carried out on real datasets show the effectiveness of the proposed approach in comparison to state-of-the-art techniques both in terms of performance and time efficiency, especially when more than one topic is present in the corpus.

1 Introduction

Anomaly Detection (AD) is a task that can address various objectives such as mining frauds (Deng and Mei, 2009), diseases (Han et al., 2021) and intrusions (Pu et al., 2021). AD takes several forms: supervised, unsupervised or semi-supervised. Unsupervised AD implies that no prior information about the dataset is provided. In this case, the solution usually consists of identifying samples that deviate in a certain way from the others among the same dataset; anomalies being, by definition, rare phenomena. Particularly, anomalies in a textual dataset can be defined as samples having an atypical vocabulary (lexical anomaly) or a deviating global meaning (semantic anomaly). Identifying abnormalities in textual data can be very useful in many industrial use-cases. A good example is the detection of non-eligible and/or fraudulent course contents in the public French plateform MonCompteFormation[1] where millions of course sessions are available with no possibility of controlling training organizations in a supervised fashion (using labeled data). Hence, to assess the effectiveness of our approach, we rely on an external labeled dataset that closely relates to course contents and that is dedicated to course certifications. The dataset is described in Appendix A. In addition to the difficulty of mining anomalies in short-text corpora of varying sizes, we also have an important computational cost constraint that is also addressed by the proposed solution.

Les lignes de commande Linux pour débutants	inlier
Formation: Introduction au Shell Bash	inlier
Administration système Unix pour les nuls	inlier
Apprendre à utiliser le terminal Ubuntu/Debian	inlier
Formation en Espagnol pour débutants	outlier

Table 1: Example illustrating the importance of semantic representations in a small corpus of short texts.

Capturing the semantics of a given text is usually performed using Word Embeddings, which consist in representing a word or a piece of text by a fixed-size vector, supposed to detain its meaning. Several word embedding techniques are available such as word2vec (Mikolov et al., 2013), GloVe (Pennington et al., 2014), fastText (Bojanowski et al., 2017), BERT (Devlin et al., 2019), DistilBERT (Sanh et al., 2019), etc. Each of the mentioned works provides ready-to-use models that are pre-trained on very large corpora and intended to be general for a given language and suitable for several NLP downstream tasks. Indeed, relying on such pre-trained models has proved efficient in several tasks (Kim, 2014; Das et al., 2017) and is particularly useful when dealing with small corpora (Buechel et al., 2018). If we consider the

[1]https://www.moncompteformation.gouv.fr/ developed by « Caisse des Dépôts et Consignations » (CDC)

Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 1392–1403
May 2-6, 2023

small corpus given in Table 1, we can observe that the inlier samples (training titles about Linux shell programming) do not have any words in common. Thus, based only on the syntactic information of the samples, it would be impossible to isolate the outlier (about learning Spanish), even though it is the only one that does not have anything to do with shell programming. This can easily occur when dealing with short-text corpora, especially when the number of samples is not sufficient to learn the different syntactic variants of a word or a concept.

Depending on the data type and the assumptions that can be made, the definition of an outlier may differ, and the choice of the model is crucial, especially in an unsupervised context (Aggarwal, 2017). In this paper, we propose a probabilistic AD approach based on Mixture Models, that effectively identifies the most deviating samples in short-text datasets, even in the case where several topics are present in the inlier class. We also show the effectiveness of using the knowledge learned by word embedding models in capturing the underlying semantics of short texts and efficiently identifying outliers. The main contributions of this paper are:

- We address the challenge of mining anomalies in short texts written in French
- We tackle the classical one-class inlier scheme and also a more challenging multi-class inlier setting.
- We propose an effective and efficient anomaly detection approach that outperforms previously proposed AD techniques.

2 Related Work

Anomaly detection is an active research area, and a large number of approaches are proposed in several application domains. Specifically, our work relates to unsupervised AD for text and clustering-based AD. Unsupervised AD is gaining more and more interest in research due to the constant growth of data volumes while labeling data samples is not getting any cheaper. One of the most important family of methods contains reconstruction-based approaches that assume that a well-generalizing model would struggle at compressing rare anomalous samples. This kind of approaches include linear models such as Robust PCA (Kang et al., 2015) and deep autoencoder models based on convolutional networks (Oza and Patel, 2019), recurrent networks (Hsieh et al., 2019), etc.

2.1 Clustering-based AD

In the clustering-based AD approaches, anomalies are generally seen as data samples that present a lower adhesion to the underlying groups. Several two-phase approaches have been proposed and consist in using a clustering algorithm such as DBSCAN (Sheridan et al.), K-means (Deng and Mei, 2009) and Affinity Propagation (Marcos Alvarez et al., 2013), then compute an anomaly score from the obtained clustering partition. Similarly in (Mahadevan et al., 2010), to detect temporal anomalies in videos, inlier behaviors are modeled as a mixture of Gaussian distributions. A deep GMM-based approach called DAGMM is proposed in (Zong et al., 2018) to detect outliers in numerical data, where the input data are compressed into a lower-dimensional space using an autoencoder and then fed into a GMM component. The autoencoder's reconstruction loss and the log-likelihood of the GMM component are optimized jointly, without performing any pre-training phase.

2.2 AD in text data

Unlike images, time series, and numerical data, relatively few AD studies are dedicated to textual data. Document-term matrix representations (also called sparse bag-of-words) have previously been used in (Kannan et al.) to perform AD based on Nonnegative Matrix Factorisation (NMF) and isolate an outlier matrix, used to compute the anomaly scores. Sparse representations are also used by Manevitz and Yousef (2001) as input to a One-Class Support Vector Machine (OC-SVM) (Schölkopf et al., 2001) and later to a shallow autoencoder (Manevitz and Yousef, 2007). Word embeddings like word2vec are used for AD in (Zhuang et al., 2017) along with a von Mises Fisher (vMF) mixture model where more general words are penalized when computing the overall outlierness score of a given document. Pre-trained fastText word vectors are used in (Ruff et al., 2019) as the embedding layer of a multi-head attention network to perform anomaly detection as a one-class classification task. Recently, a deep end-to-end approach has been proposed by Manolache et al. (2021) that does not use any knowledge transfer. The authors use the transformer architecture of ELECTRA (Clark et al., 2020) that contains two adversarial components: a generator and a discriminator. The model is trained from scratch on a given dataset by optimizing a loss function based on token replacement.

2.3 Semantic text representations

Tremendous advances in various NLP tasks have been made in recent years thanks to dense vector representations of words and text sequences. Static word embeddings like word2vec (Mikolov et al., 2013), GloVe (Pennington et al., 2014) and fastText (Bojanowski et al., 2017) provide one unique dense representation for each word whereas contextual word embedding models like ELMo (Peters et al., 2018), BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) provide word representations that depend on the surrounding context. Contextual word embedding models are based on deep neural networks, which makes them resource-intensive and difficult to use in some industrial contexts. Both kinds of word embeddings have proved effective in several unsupervised downstream tasks like semantic textual similarity (Arora et al., 2017; Ranasinghe et al., 2019), clustering (Ait-Saada et al., 2021; Boutalbi et al., 2022) and anomaly detection (Zhuang et al., 2017).

3 Gaussian Mixture Models

Given a corpus $\mathcal{D}$ of n short texts $(d_1, \ldots, d_n)$, we represent each sample by a fixed size vector, thus obtaining a matrix $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ of size $n \times m$. To tackle the AD problem, we postulate that the samples follow a mixture of distributions, from which the anomalous samples deviate.

Admittedly, the family of t-distributions provides a heavy-tailed alternative to the gaussian family for anomaly detection. However, as pointed out by (Yuan and Huang, 2009), although useful from modeling perspective, the practical use of multivariate t-distribution is often limited by the difficulty in parameter estimation, particularly so for high dimensional data. Note that, in our proposal, the consideration of the time consumption is important. Therefore, considering a t mixture model leads to estimate a supplementary parameter (in addition to the estimation of vector means and covariance matrices) that is the degree of freedom of each component. Moreover, since we suggest to consider an ensemble method allowing to combine results by varying the number of components (cf. Section 3.1), we would therefore increase yet the computation time for estimation of the parameters. However, Gaussian Mixture Model (GMM)-based approaches, more parsimonious than t-mixture model, have shown their effectiveness in anomaly detection, such as DAGMM (Zong et al., 2018). For these reasons we retain GMM to address our purpose.

In a finite GMM, the data $(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ are taken to constitute a sample of n independent instances of a random variable $\mathbf{X}$ in $\mathbb{R}^m$. Density can be expressed as:

$$f(\mathbf{x}_i; \Theta) = \sum_{k=1}^{g} \pi_k \varphi_k(\mathbf{x}_i | \mu_k, \Sigma_k), \forall i \in \{1, \ldots, n\}$$

where $\Theta = (\pi_1, \ldots, \pi_g, \mu_1, \ldots, \mu_g, \Sigma_1, \ldots, \Sigma_g)$, $\varphi_k(\mathbf{x}_i | \mu, \Sigma_k)$ is the kth component density for observation $\mathbf{x}_i$ with parameters (μ_k, Σ_k), $(\pi_1, \ldots, \pi_{g-1})$ are the mixing weights or probabilities (such that $\pi_k > 0$, $\sum_{k=1}^{g} \pi_k = 1$) and g is the number of mixture components. Thus, clusters are ellipsoidal, centered at the mean vector μ_k, and with other geometric features, such as volume, shape and orientation, determined by the covariance matrix Σ_k (Banfield and Raftery, 1993; Celeux and Govaert, 1995). To estimate Θ we rely on the maximisation of the log-likelihood given by:

$$L(\mathbf{X}; \Theta) = \sum_{i=1}^{n} \log \left(\sum_{k=1}^{g} \pi_k \varphi_k(\mathbf{x}_i | \mu_k, \Sigma_k) \right).$$

The maximization is commonly performed by Expectation-Maximization (Dempster et al., 1977); an iterative algorithm based on the maximization of the conditional expectation of the complete data log-likelihood given Θ':

$$Q(\Theta | \Theta') = \sum_i \sum_k s_{ik} \log(\pi_k \varphi_k(\mathbf{x}_i | \mu_k, \Sigma_k))$$

where $s_{ik} \propto \pi_k \varphi_\ell(\mathbf{x}_i | \mu_k, \Sigma_k)$ are the posterior probabilities.

In real terms, the algorithm is broken down into two steps (E-M steps) and the unknown parameters of Θ are updated thanks to the previously computed probabilities. For each component k, we have

$$\pi_k = \frac{\sum_i s_{ik}}{n} \quad \mu_k = \frac{\sum_i s_{ik} x_{ij}}{\sum_i s_{ik}}$$

$$\text{and } \Sigma_k = \frac{\sum_i s_{ik} (\mathbf{x}_i - \mu_k)^\top (\mathbf{x}_i - \mu_k)}{\sum_i s_{ik}}.$$

The procedure used to identify anomalies is described in Algorithm 1. It takes as input a set of short texts and returns the ones that are the most likely to constitute an anomaly. The maximum density as normality score denotes the confidence

Algorithm 1: AD with GMM

Input: $\mathcal{D} = \{d_1, \dots d_n\}$, g the number of components, $\mathcal{M}$ an embedding module, α the desired number of output samples;
$\mathbf{x}_i \leftarrow \mathcal{M}(d_i)$, $i = 1, \dots n$;
$\mathbf{X} \leftarrow (\mathbf{x}_1, \dots, \mathbf{x}_n)$;
Initialize Θ' from a partition obtained with k-means
repeat
  E-step: Compute $Q(\Theta|\Theta')$;
  M-step: Update π_k, μ_k and Σ_k;
until *Convergence*;
$s_i \leftarrow -\max_k(s_{ik}), k = 1, \dots, g$;
$\mathbf{s} \leftarrow (s_i, \dots, s_n)$;
$\mathbf{r} \leftarrow$argsort$(\mathbf{s})$;
return d_j, $j = \mathbf{r}_1, \dots, \mathbf{r}_\alpha$;

of the assignment. Multiplied by -1, it denotes the uncertainty of the assignment and is similar to using the entropy of $\mathbf{s}$ since $\sum_k s_{ik} = 1$. The number of returned text samples depends on the user's needs and is specified by the cutoff parameter α. In the evaluation section, we evaluate the AD performance with every possible value of α using the AUROC score.

3.1 Proposed solution for multi-class inliers

In the standard setting of AD where we consider one large inlier class, we set the number of components to its smallest possible value $g = 2$, which provides satisfactory results. In this study, we also consider the more challenging scenario where several underlying topics are present in the dataset. In this context, we make the distinction between extreme values and outliers (Aggarwal, 2017) as shown in Figure 1. A Gaussian mixture model would not have any difficulty in spotting both types of outliers since it is capable of modeling clusters of different shapes. Furthermore, we expect GMM to show good results in the multi-class context, since one of its fundamental assumptions is the multiplicity of inherent distributions among the data samples. However, this property requires to know the number of components in advance, which is not always possible in real life.

To address this issue, we propose to use GMME, an ensemble of several models, obtained with different values of g. To this end, we use Algorithm 1 with varying $g_k \in \mathcal{G}$ and combine the output scores

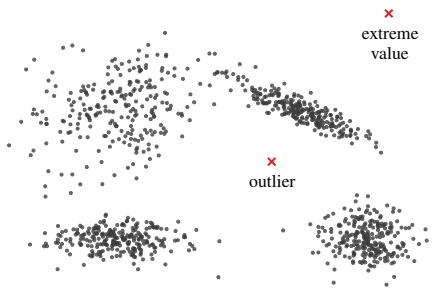

Figure 1: Difference between outlier and extreme value. This example illustrates the benefit of the clustering-based AD approaches in general and Gaussian mixture models in particular.

$\mathbf{s}^{(g_k)}$ as follows:

$$\mathbf{e}_i = -\sum_k \text{rank}(s_i^{(g_k)}).$$

The intuition behind using an ensemble approach is to make each of the models separate the dataset into clusters in a different way and assign an anomaly score according to the formed clusters. Combining those different anomaly scores leads to a more robust and meaningful overall score, even when the optimal number of clusters is not included in $\mathcal{G}$. This is corroborated by the empirical study conducted in Section 4.5.

4 Experimental Study

To assess the effectiveness of our approach and compare it to state-of-the-art, we conduct a set of experiments on real datasets and discuss the results in this section.

4.1 Datasets

We run our AD experiments on three datasets described in Table 2. MLSUM (Scialom et al., 2020) and COVID-news (Cortal, 2022) are both news datasets from which we extract the title to constitute our short-text corpora. RNCP is a dataset we built from an official French repository that lists training certifications (cf. Appendix A).

Dataset	Classes	Smallest	Largest	Medial
RNCP	16	763	9,510	3,101
COVID	9	236	3,235	1,270
MLSUM	10	2,573	26,024	13,054

Table 2: Datasets' description. The sizes correspond to the whole set of samples (training and test set).

Given a classification dataset, we first remove the classes that are too small to constitute an AD

dataset, thus obtaining ℓ classes. We then derive ℓ sets of samples in which there is one inlier (majority) class and a certain rate r of outliers picked randomly from the other classes.

4.2 Experimental settings

In order to empirically evaluate our approach and compare it to other AD techniques, we rely on the Area Under the Receiver Operating Curve (AUROC), originally used as a metric in the classification task. In our case, it takes as input the anomaly scores as well as the ground truth labels and determines at what extent it is possible to accurately identify outliers using the anomaly score. It is equivalent to evaluating the performance of AD at every possible value of α in Algorithm 1. For each AD approach, we compute the AUROC on the test set over 5 different initializations, except for OC-SVM that is a deterministic model.

In our study, we discard the case where we train the model on the majority class as a one-class classification task (Ruff et al., 2019; Manolache et al., 2021; Manevitz and Yousef, 2001), for it is not a realistic scenario since one rarely has access to a large enough amount of inlier-only labeled samples in real life. Thus, we consider in this study a fully unsupervised scenario, where no labels are available and both inlier and outlier samples are present in the training set. To this end, we contaminate both sets with up to $r = 10\%$ of outliers as in (Manolache et al., 2021). The set of labels is used only during the evaluation phase.

Text representation. Given a raw corpus $\mathcal{D}$ of short texts, we first perform a minimal preprocessing that consists in removing stop words and lowercasing the input text. Then, we use a pretrained fastText model (Bojanowski et al., 2017) to represent texts by fixed-size vectors and build the $\mathbf{X}$ matrix. The model is trained on French Wikipedia and represents each word by a vector of size $m = 300$. Using those word vectors, we represent a text sequence by the arithmetic mean of its tokens' representations as in (Ranasinghe et al., 2019; Arora et al., 2017). We show that this way of representing text sequences is well suited to shorts texts and is very beneficial in capturing text semantics for AD. One noticeable advantage of fastText is its ability to represent out-of-vocabulary words thanks to sub-word embeddings. Another advantage of fastText is that pre-trained word representations are provided in a wide range of languages[2]. We do not use contextual word embeddings to represent text sequences since they significantly increase the computational cost and do not seem to bring any performance gain in the AD task (Ruff et al., 2019).

Baselines. We compare our approach to other AD techniques: OC-SVM (Schölkopf et al., 2001), AE, DAGMM (Zong et al., 2018) and DATE (Manolache et al., 2021). For OC-SVM, AE and DAGMM, we use as input the same matrix $\mathbf{X}$ as for GMM. Concerning OC-SVM, we set $\nu = 0.05$, which is the value that presents the best results among $\{0.05, 0.1, 0.2, 0.5\}$ by far. For the autoencoder (AE) we train a model with three encoder layers and three decoder layers of size 256, 128 and 64, a learning rate of 0.001 and a weight decay of 10^{-8}. Concerning DAGMM, the authors choose the parameters relating to the architecture of the neural network according to the dataset and do not provide a method to reproduce this choice. This way of configuring the model is not suitable for the unsupervised case, in which no tuning of the hyperparameters is possible. We therefore opt for a standard architecture in decreasing powers of 2 starting from $m = 300$ (i.e. $256, 128, \dots$). We choose $m' = 5$ as the encoding dimension because it gives the best overall performance. For DATE, we use the same parameters as used in the original paper for AG-news (Manolache et al., 2021).

Hyperparameter tuning. Our present work falls within the context of unsupervised learning, where we are not supposed to have access to ground truth labels. In this respect, it is not feasible in real life to tune the hyperparameters of a given model to fit unlabeled data, since the performance score is simply impossible to compute. We therefore consider hyperparameter tuning in this context unrealistic and possibly leading to hiding instabilities that can neither be detected nor fixed by practitioners. Hence, robustness and insensitivity to parameters are key in the unsupervised setting. For this reason, regarding the baselines, we use the recommended parameters provided in the original works when available, in order to reproduce real-world conditions. When no recommended parameters are provided, we use the ones that maximize the overall performance even though it does not play in favor of our proposed approach which does not require any parameter tuning. By doing so, we guaran-

[2] https://fasttext.cc/

	Inlier	OC-SVM	AE	DAGMM	DATE	GMM
RNCP	enviro.	62.2	62.1	52.8	65.8	**68.4**
	défens.	58.0	46.2	52.5	62.2	**74.3**
	intell.	66.7	65.4	59.1	79.0	**80.1**
	recher.	68.6	64.5	58.0	75.3	**77.7**
	nautis.	64.9	62.9	53.0	68.2	**73.1**
	aérono.	63.7	59.4	49.8	76.1	**78.0**
	sécuri.	59.8	54.7	51.3	75.0	**80.0**
	multim.	57.8	56.3	51.3	64.5	**71.2**
	humani.	56.7	59.6	53.4	69.0	**72.1**
	nucléa.	65.1	58.7	59.9	74.9	**75.1**
	enfanc.	61.0	60.1	65.0	72.1	**78.9**
	saison.	50.5	47.5	50.8	48.8	**74.9**
	assist.	45.1	50.7	56.4	52.5	**63.2**
	sport	51.1	56.1	46.8	66.3	**73.3**
	ingéni.	66.9	60.1	57.2	**75.3**	74.2
	sans d.	45.4	36.9	53.7	39.3	**59.9**
COVID	cultur.	49.7	40.1	43.4	40.3	**53.7**
	enviro.	56.9	55.0	50.2	55.2	**66.2**
	intern.	62.0	53.7	53.1	62.4	**65.1**
	people.	63.9	47.4	49.7	51.8	**64.0**
	politi.	67.2	60.1	51.6	68.1	**79.2**
	scienc.	55.5	38.3	60.6	64.5	**66.6**
	sociét.	48.9	47.5	55.3	**57.0**	55.1
	sport	68.5	35.6	55.4	52.6	**69.0**
	économ.	52.7	49.4	50.7	54.0	**59.4**
MLSUM	afriqu.	70.0	50.5	62.0	74.3	**75.5**
	police.	78.3	74.4	52.9	77.0	**82.9**
	politi.	70.3	60.3	49.6	74.6	**75.7**
	scienc.	53.1	35.5	49.6	**71.0**	55.2
	societ.	73.0	73.0	48.9	75.4	**79.0**
	sante	78.7	56.2	55.2	86.5	**87.5**
	argent.	47.7	36.4	51.7	**82.2**	63.4
	livres.	**71.2**	65.8	48.4	58.1	65.9
	cultur.	**59.5**	49.8	52.4	45.8	51.4
	sport	74.9	56.5	45.2	69.2	**81.7**

Table 3: AUC scores obtained with anomaly $r = 10\%$. The **bold** numbers correspond to the best score in each row and the underlined numbers are for the second best performance score.

tee a fair evaluation of our proposal and show its robustness in an unsupervised context.

4.3 Results with one-class inliers

The obtained performance is given in Table 3 with an anomaly rate $r = 10\%$. We first observe the effectiveness of GMM on the three datasets, in comparison to all of the baselines, offering the best AUROC in most of the cases. Furthermore, the state-of-the-art DATE shows its limits on short texts and presents competitive but poorer results in comparison to GMM. OC-SVM is competitive on short texts in comparison to DATE but presents poorer overall performance. AE is the model that provides the lowest AUROC values, right after DAGMM. This might be due to the fact that the encoding (or embedding) step of the encoder (for both AE and DAGMM) is performed beforehand using pre-trained word embeddings and becomes pointless when applied to this kind of representation.

The results obtained with different values of contamination rate r are summarized by Critical Difference (CD) diagrams in Figure 2. The aim of CD diagrams (Demšar, 2006) is to visualize the performance ranks of each approach over the different datasets. If we take the example of $r = 10\%$, the CD diagram summarizes the scores given in Table 3. It depicts the average rank of each method and the bold line corresponds to the critical difference, based on the post-hoc Nemenyi test (Nemenyi, 1963). Note that the presence of a higher number of outliers increases the difficulty of the AD task and decreases the overall performance scores. We can observe that the more we inject anomalies in the training set, the more GMM gets competitive against the other approaches in terms of AUROC score. This shows GMM's robustness to outliers and its generalization capabilities while other techniques tend to overfit in the presence of noise in the training set.

Also, it is worth noting that GMME yields similar results in comparison to GMM, which makes it a universal solution, well suited to AD even in the one-class scenario.

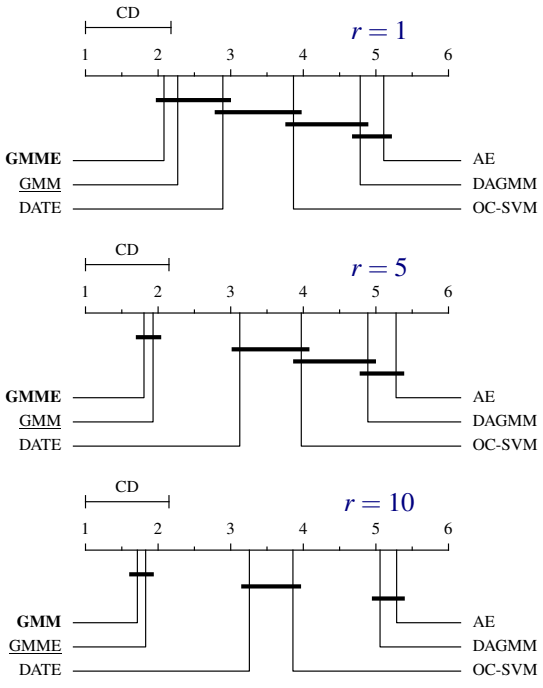

Figure 2: CD plots from the Nemenyi test over different datasets. This graphic summarizes the rank of each approach with different contamination rates r.

Comparison with VMF-Q. To make a fair comparison between GMM and VMF-Q (Zhuang et al., 2017), we reproduce the exact same setting for the two approaches. VMF-Q is based on the von Mises-Fisher distribution that relies on a Bessel function to estimate the parameter κ. The model is originally trained using embeddings of size $m = 200$, but encounters numerical difficulties with higher dimensions, due to the approximations made by the Bessel function that depends, inter alia, on m. We hence use, for both GMM and VMF-Q, another pre-trained model of size $m = 200$, that is provided by Fauconnier (2015). The gain of performance from VMF-Q to GMM is summarized in Figure 3. We can observe a clear advantage of GMM in comparison to VMF-Q on the three datasets, especially on MLSUM, where GMM outperforms VMF-Q on all the subsets. We also report poorer overall results using word2vec with $m = 200$ in comparison to the fastText model we use in the rest of our experiments.

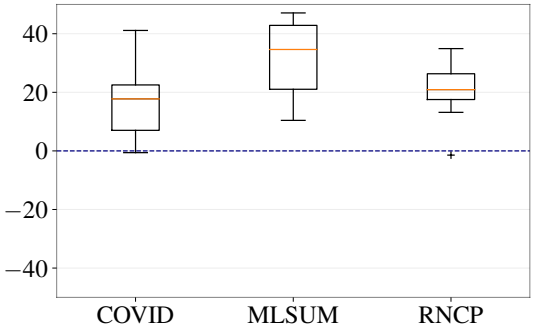

Figure 3: Gain of performance between GMM and VMF-Q using word2vec embeddings of size $m = 200$. Positive values give an advantage to GMM.

4.4 Performance and data size

Figure 4 shows the gain of performance from baselines to GMM according to the size of the datasets. We see that the percentage of improvement is greater on small datasets but remains positive on large datasets. Note that detecting anomalies can be trickier on small datasets, especially when dealing with short texts (cf. Table 1) which makes GMM a good solution to tackle this difficulty.

4.5 Multi-class inliers

We investigated in the previous sections the detection of semantic anomalies in a dataset classically composed of one unique class. In this section, we consider a dataset with several underlying topics and identify the samples that do not belong to any

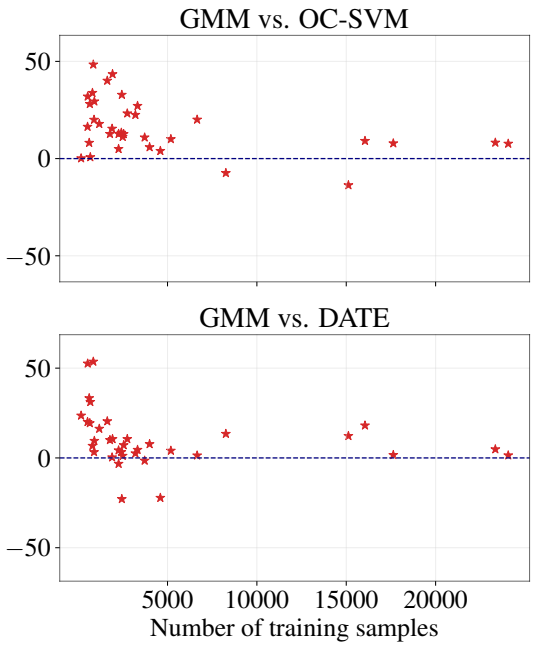

Figure 4: Percentage of improvement of GMM in comparison to baselines w.r.t. to the train set's size. Positive values give an advantage to GMM.

of them. To evaluate our approach in such a context, we create datasets as described in Section 4.1 but this time, by combining $\hat{g}$ random classes to form one inlier class then inject anomalous samples with a rate $r = 10\%$. We then proceed similarly to identify anomalies and validate the obtained results. Note that we make sure in our experiments to put aside enough "anomalous" classes so that we have a sufficient diversity of anomalies and avoid forming an additional cluster with the anomalous samples. To this end, we limit $\hat{g}$ according to the available classes in the dataset. We use the obtained datasets to assess the performance of GMME (cf. Section 3) and compare it to GMM and two other baselines: DBSCAN and DATE. DBSCAN natively deals with outliers and considers the samples that cannot be assigned to any existing cluster as anomalies (they are assigned the -1 label). We use this information to determine whether a text sample is an anomaly. We set the parameters $\epsilon = 1$ and min_samples $= 3$ which yielded better results than the default values available in the scikit-learn implementation.

Table 4 shows the performance obtained by GMME with values of $g_k \in \mathcal{G} = \{2, 3, 4\}$. We first observe that the ensemble approach further improves the performance of the simple GMM, which still performs better in comparison to the other baselines. DBSCAN presents the poorest results after

	# inlier classes	DBSCAN	DATE	GMM	GMME
RNCP	2	63.4	75.6	88.6	**89.2**
	3	62.8	70.5	82.4	**83.9**
	4	64.6	77.6	80.2	**82.0**
	5	65.7	63.5	**81.6**	80.7
	6	61.7	56.7	**65.5**	64.5
	7	58.4	48.8	60.6	**62.6**
	8	60.4	56.8	63.2	**63.4**
COVID	2	51.5	59.3	**65.2**	63.8
	3	50.6	51.3	54.1	**54.2**
	4	50.7	53.5	56.2	**56.3**
	5	50.0	51.1	52.6	**52.7**
MLSUM	2	50.5	70.0	74.2	**74.6**
	3	50.2	49.6	**55.1**	54.4
	4	50.6	47.0	55.0	**55.5**
	5	50.7	45.7	53.1	**53.8**

Table 4: Comparison of DBSCAN, DATE, GMM and GMME approaches: AUROC scores obtained with multiple classes as inliers.

DATE, which is less competitive in the multi-class context, especially on the RNCP dataset. Thus, GMME is the most competitive approach in the multi-topic scenario, even when the real number of clusters is not included in $\mathcal{G}$.

4.6 Computation time analysis

Table 5 contains the execution time of both the training and evaluation steps. It is estimated over three different runs, on the three-class datasets used in Section 4.5. The sizes of the training sets are 1034, 6342, 46253 and the test sets are of size 444, 2718, 5837 for RNCP, COVID-news and MLSUM respectively. The experiments on DATE are performed on an NVIDIA RTX2070 GPU.

We first notice that GMM is the fastest approach both during the training and evaluation phase. It is followed by GMME that is relatively quick, especially during evaluation. GMM scales better with an increasing number of samples in comparison to OC-SVM that takes more than four times as much time to train. DATE is the approach that takes the most time to train and evaluate, which is due to its deep architecture. The computational time of DATE can be partially amortized with a more powerful GPU but can still represent an impediment, especially in an industrial context.

We also observe that the computation time of $\mathbf{X}$ does not depend much on the size of the dataset, simply because the task that takes the longest is loading the model from memory. Hence, this way

		1-SVM	GMM	GMME	DATE
RNCP	$\mathbf{X}$	3.7			-
	train	3e-01	3e-01	1.2	78.49
	eval	1e-01	3e-03	3e-02	2.0
COVID	$\mathbf{X}$	3.7			-
	train	5.7	3.5	16.2	512.97
	eval	1.8	2e-02	8e-02	14.1
MLSUM	$\mathbf{X}$	3.9			-
	train	450.6	10.0	94.8	3639
	eval	33.0	4e-02	2e-01	30.2

Table 5: Execution time in seconds. The row $\mathbf{X}$ corresponds to the computation time of the embedding matrix $\mathbf{X}$ using fastText. "train" stands for the training time and "eval" for the evaluation time.

of representing text scales well, especially when used along with GMM or GMME.

4.7 Improve the results with Transformers

In the previous experiments, we relied on fastText static embeddings in order to show that is it possible to perform effective and fast anomaly detection without having access to important computational resources. In this section, we show that it is possible to achieve even better results using Transformer representations. For our purpose, we use CamemBERT (Martin et al., 2020) and FlauBERT (Le et al., 2020), both trained on French corpora.

CamemBERT and FlauBERT are both based on the RoBERTa variant of Transformer language models (Liu et al., 2019). While FlauBERT uses the exact same objective and tokenization process as in Liu et al. (2019), plus a French-specific preprocessing, CamemBERT makes use of the SentencePiece tokenizer and the whole-word masking strategy that consists in masking words instead of sub-word tokens. Another major difference between the two models is the training set of data. CamemBERT uses the French part of OSCAR, a large multilingual corpus extracted from Common Crawl, while FlauBERT is trained on a set of 24 corpora.

Table 6 contains the results obtained using several input representations. We recall the results obtained by DATE for comparison purposes since it is also based on a Transformer architecture. We first observe the significant difference in performance between CamemBERT and FlauBERT with a clear advantage for CamemBERT in both its base and large versions. The large variant of Camem-

Dataset	# Inlier classes	DATE	GMME					
			fastText	flauB_6	flauB_{12}	flauB_{24}	camB_{12}	camB_{24}
RNCP	2	75.58	**89.22**	87.02	59.78	71.38	83.82	88.44
	3	70.48	**83.90**	76.10	64.80	63.38	76.98	82.14
	4	77.58	82.00	79.14	70.82	66.06	80.82	**84.82**
	5	63.50	80.68	69.76	47.78	69.80	77.60	**83.54**
	6	56.74	64.46	60.48	53.40	60.48	64.60	**66.64**
	7	48.78	62.60	64.68	67.24	66.42	67.84	**68.66**
	8	56.82	63.44	53.18	47.28	59.92	65.16	**67.00**
COVID	2	59.28	63.82	62.34	56.38	56.56	**65.82**	63.32
	3	51.32	54.20	54.10	47.28	54.86	**56.00**	55.48
	4	53.46	56.30	55.40	49.70	51.80	58.00	**58.44**
	5	51.08	52.70	**56.28**	49.02	50.48	52.90	52.12
MLSUM	2	70.00	74.60	68.80	61.00	55.00	79.60	**85.10**
	3	49.60	54.44	59.70	47.80	56.50	66.44	**72.74**
	4	47.02	55.52	58.30	39.00	54.70	65.02	**70.18**
	5	45.70	53.84	58.80	42.30	56.02	65.44	**71.08**

Table 6: Comparison between different embedding representations in terms of AUC score with anomaly rate $r = 10\%$. The **bold** numbers correspond to the best score in each row and the underlined numbers are for the second-best performance score. flauB and camB stand respectively for FlauBERT and CamemBERT and the subscript is for the number of layers, which indicates the model size (6 for small, 12 for base and 24 for large).

BERT achieves the best overall results, especially on MLSUM. CamemBERT-base yields competitive results, achieving good AUROC scores on the COVID dataset. We also notice that, in all cases, the two-phased approach that consists in computing Transformer representations and then use an ensemble of model-based clustering models (GMME) is more efficient than the end-to-end Transformer-based approach (DATE). It is worth noting that fastText embeddings remain competitive and surpass the three FlauBERT models. It is therefore a very good alternative in real-time use cases when computational speed is a critical issue.

5 Conclusion

This paper addresses semantic anomaly detection in short texts with an additional constraint in time efficiency. In addition to the classical framework where one class is used as the inlier class, we also consider the scenario where several underlying subgroups are present in the normal class. We see anomaly detection as a probabilistic clustering problem, in which we learn a Gaussian mixture model and consider the low posterior probability samples as belonging to none of the modeled clusters and more likely to constitute outliers. This uncertainty score proved effective with different numbers of subgroups. In the multi-class setting, we propose GMME, an ensemble approach that improves the performance of GMM when several topics are present in the inlier class. The two approaches outperform state-of-the-art anomaly detection techniques in both scenarios, with an impressively low computation time.

In our proposal, we rely on the Gaussian Mixture model for its flexibility. This choice is motivated by the presence of the proportions π_k of each cluster and the spectral decomposition of the covariance matrix Σ_k taking into account the volume, shape, and orientation of each cluster (as depicted in Figure 1). The characteristics of the clusters should not be overlooked when tackling the problem of anomaly detection through a clustering approach. Furthermore, note that our approach can be extended to latent block models, devoted to co-clustering, which may constitute an interesting and promising future path of research.

6 Limitations

This paper deals exclusively with French text corpora to answer a specific industrial need. However, we are confident about the fact that this work can easily be extended to other languages, especially English, for which more data and pre-trained models are available. This would constitute an interesting trajectory for future work.

References

Charu C. Aggarwal. 2017. An introduction to outlier analysis. In *Outlier analysis*, pages 1–34. Springer.

Mira Ait-Saada, François Role, and Mohamed Nadif. 2021. How to leverage a multi-layered transformer language model for text clustering: An ensemble approach. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, page 2837–2841.

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2017. A simple but tough-to-beat baseline for sentence embeddings. In *International Conference on Learning Representations*.

Jeffrey D. Banfield and Adrian E. Raftery. 1993. Model-based gaussian and non-gaussian clustering. *Biometrics*, 49(3):803–821.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Rafika Boutalbi, Mira Ait-Saada, Anastasiia Iurshina, Steffen Staab, and Mohamed Nadif. 2022. Tensor-based graph modularity for text data clustering. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 2227–2231.

Sven Buechel, João Sedoc, H. Andrew Schwartz, and Lyle H. Ungar. 2018. Learning neural emotion analysis from 100 observations: The surprising effectiveness of pre-trained word representations. *CoRR*, abs/1810.10949.

Gilles Celeux and Gérard Govaert. 1995. Gaussian parsimonious clustering models. *Pattern Recognition*, 28(5):781–793.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Gustave Cortal. 2022. Covid-19 - french news dataset.

Arjun Das, Debasis Ganguly, and Utpal Garain. 2017. Named entity recognition with word embeddings and wikipedia categories for a low-resource language. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 16(3).

A. P. Dempster, N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22.

Janez Demšar. 2006. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7(1):1–30.

Qingshan Deng and Guoping Mei. 2009. Combining self-organizing map and k-means clustering for detecting fraudulent financial statements. In *2009 IEEE International Conference on Granular Computing*, pages 126–131.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pages 4171–4186.

Jean-Philippe Fauconnier. 2015. French word embeddings.

Changhee Han, Leonardo Rundo, Kohei Murao, Tomoyuki Noguchi, Yuki Shimahara, Zoltán Ádám Milacski, Saori Koshino, Evis Sala, Hideki Nakayama, and Shin'ichi Satoh. 2021. Madgan: Unsupervised medical anomaly detection gan using multiple adjacent brain mri slice reconstruction. *BMC bioinformatics*, 22(2):1–20.

Ruei-Jie Hsieh, Jerry Chou, and Chih-Hsiang Ho. 2019. Unsupervised online anomaly detection on multivariate sensing time series data for smart manufacturing. In *2019 IEEE 12th Conference on Service-Oriented Computing and Applications (SOCA)*, pages 90–97.

Zhao Kang, Chong Peng, and Qiang Cheng. 2015. Robust pca via nonconvex rank approximation. In *2015 IEEE International Conference on Data Mining*, pages 211–220.

Ramakrishnan Kannan, Hyenkyun Woo, Charu C. Aggarwal, and Haesun Park. *Outlier Detection for Text Data*, pages 489–497.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoit Crabbé, Laurent Besacier, and Didier Schwab. 2020. FlauBERT: Unsupervised language model pre-training for French. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2479–2490, Marseille, France.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Vijay Mahadevan, Weixin Li, Viral Bhalodia, and Nuno Vasconcelos. 2010. Anomaly detection in crowded scenes. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1975–1981.

Larry Manevitz and Malik Yousef. 2007. One-class document classification via neural networks. *Neurocomputing*, 70(7):1466–1481. Advances in Computational Intelligence and Learning.

Larry M Manevitz and Malik Yousef. 2001. One-class svms for document classification. *Journal of machine Learning research*, 2(Dec):139–154.

Andrei Manolache, Florin Brad, and Elena Burceanu. 2021. DATE: Detecting anomalies in text via self-supervision of transformers. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 267–277.

Alejandro Marcos Alvarez, Makoto Yamada, Akisato Kimura, and Tomoharu Iwata. 2013. Clustering-based anomaly detection in multi-view data. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*, page 1545–1548.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. CamemBERT: a tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, volume 26.

Peter Bjorn Nemenyi. 1963. *Distribution-free multiple comparisons.* Princeton University.

Poojan Oza and Vishal M. Patel. 2019. One-class convolutional neural network. *IEEE Signal Processing Letters*, 26(2):277–281.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237.

Guo Pu, Lijuan Wang, Jun Shen, and Fang Dong. 2021. A hybrid unsupervised clustering-based anomaly detection method. *Tsinghua Science and Technology*, 26(2):146–153.

Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2019. Enhancing unsupervised sentence similarity methods with deep contextualised word representations. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 994–1003, Varna, Bulgaria. INCOMA Ltd.

Lukas Ruff, Yury Zemlyanskiy, Robert Vandermeulen, Thomas Schnake, and Marius Kloft. 2019. Self-attentive, multi-context one-class classification for unsupervised anomaly detection on text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4061–4071.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108.

Bernhard Schölkopf, John C. Platt, John Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. 2001. Estimating the Support of a High-Dimensional Distribution. *Neural Computation*, 13(7):1443–1471.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, and Jacopo Staiano. 2020. MLSUM: The multilingual summarization corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8051–8067.

Kevin Sheridan, Tejas G. Puranik, Eugene Mangortey, Olivia J. Pinon-Fischer, Michelle Kirby, and Dimitri N. Mavris. *An Application of DBSCAN Clustering for Flight Anomaly Detection During the Approach Phase*.

Ming Yuan and Jianhua Z. Huang. 2009. Regularized parameter estimation of high dimensional t distribution. *Journal of Statistical Planning and Inference*, 139(7):2284–2292.

Honglei Zhuang, Chi Wang, Fangbo Tao, Lance Kaplan, and Jiawei Han. 2017. Identifying semantically deviating outlier documents. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2748–2757.

Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. 2018. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *International Conference on Learning Representations*.

A Construction of the RNCP dataset

The RNCP[3] dataset is composed of certification contents that are provided by an organization named « *France-Compétences* ». All the data used here are publicly available in `https://www.data.gouv.fr/fr/datasets/`.

In order to label data we first get the ROME codes of each certification that correspond to the

[3]Répertoire National des Certifications Professionnelles

professions related to the certification. Each ROME code is assigned to a topic in the file « Arborescence thématique » of the same repository. We use the ROME code as intermediate to have the topic (*thématique*) of each certification. We hence obtain a multi-label classification dataset.

Given a certification c_i, let $\mathcal{T}_i$ be the set of its corresponding topics. In the one-class setting with inlier class h, the label of c_i is established as follows:

$$y_i = \begin{cases} 0 & \text{if } h \in \mathcal{T}_i \\ 1 & \text{otherwise.} \end{cases}$$

In the multi-class setting with inlier classes symbolized by $\mathcal{H}$, the anomaly labels are defined as:

$$y_i = \begin{cases} 0 & \text{if } \mathcal{H} \subset \mathcal{T}_i \\ 1 & \text{if } \mathcal{H} \cap \mathcal{T}_i = \emptyset \\ \text{N/A} & \text{otherwise} \end{cases}$$

where $y_i = 1$ means c_i is categorized as an anomaly.

B AD examples with RNCP

Table 7 presents some examples of anomalies predicted on two subsets of the RNCP dataset: « *aéronotique* » (meaning aeronautics) and « *nucléaire* » (meaning nuclear). The *aéronautique* test set contains 1585 samples, 154 of which are labeled as anomalies, and the *nucléaire* set contains 1431 samples including 131 anomalies. In both cases we set α to 250 (cf. Algorithm 1). We observe in both cases that DATE has more difficulty in detecting anomalous text sequences when they are very short. For example, in the *nucléaire* set, the certification *Livreur* (meaning delivery person) does not have anything to do with the nuclear field. Yet DATE does not place it among the 250 most deviant samples and makes it the 355th anomalous sample while it is only 3rd according to GMM. This might be explained by the fact that DATE is trained from scratch and does not benefit from the semantic knowledge inherited by transfer learning.

Subset	Certification	GMM	DATE	Real
aéronotique	Sciences, Technologies, Santé - Mention : Automatique et informatique industrielle - Spécialité : Automatismes industriels	Inlier	Inlier	Inlier
	Production industrielle option ingénierie des matériaux nouveaux	Inlier	Inlier	Inlier
	Actuaire	Outlier	Inlier	Outlier
	Sciences Politiques	Outlier	Inlier	Outlier
	CQP Animateur de patinoire option hockey sur glace	Outlier	Outlier	Outlier
	Décor architectural opt. B Domaine du décor du mur	Outlier	Outlier	Outlier
nucléaire	Culture et communication Mention : Création, innovation, information numériques Spécialité : Gestion de l'information et du document Domaine : Culture et communication	Inlier	Inlier	Inlier
	Responsable d'ingénierie des systèmes d'information et de communication, option "analyse et développement", option "systèmes et réseaux" et option "télécommunications"	Inlier	Inlier	Inlier
	Livreur	Outlier	Inlier	Outlier
	Architecte d'intérieur	Outlier	Inlier	Outlier
	Responsable conception, mise en place et maintenance des installations frigorifiques et climatiques	Outlier	Outlier	Outlier
	Urbanisme et Aménagement Spécialité DYATER (Dynamiques et Aménagement des espaces, Territorialités)	Outlier	Outlier	Outlier

Table 7: Examples illustrating the difference of prediction between GMM and DATE according to the length of the text sequence, with $\alpha = 250$.